CHAPTER 15

PORTABLE/HANDHELD DESIGNS

The Universal Serial Bus was conceived as an easy-to-use desktop expansion bus and it is serving this role very well. But, as with all good technological solutions, creative engineers throughout the world and extending the applications envelope of USB. Advances in storage technologies, and the increase in speed offered by USB 2.0 with further fuel the dynamic range of USB solutions.

The original usage model of a USB I/O device was one that was easy-to-connect to a PC host. Once this connection was made, the device stayed connected and served as a “semi-permanent” resource to the PC host. Printers, scanners, keyboards and mice are good examples of this “resource” usage model. The hot-plug capability of USB allowed these resources to be reallocated and this generally improved the usefulness of a USB-based I/O device.

This chapter looks at the growing portable and handheld USB based I/O device product range. I shall use the term “casually connected” to contrast these applications from their “semi-permanent resource” brethren. Most portable and handheld devices function **without** a permanent connection with the PC host; when they **are connected** then additional capability is available. We shall look inside a variety of products and I hope to expand your ideas about what can be done to enable you to create additional unique designs. I will present a variety of solutions with focus upon the building-block nature of each design and will then direct you to a variety of additional resources that will enable you to start with a similar design and then add your value-added. As always, I start with the simplest portable design and add features throughout the chapter to result in a comprehensive Personal Data Assistant (PDA) design.

A SECURITY KEY

Figure 15.1 shows a security key from Rainbow Technologies and additionally shows a conceptual block diagram of the device (this is **not** Rainbow's actual implementation).

Figure 15.1 The iKey* identity token from Rainbow Technologies

Rainbow's security key is designed to fit on your key ring or in your appointment book – you carry it with you at all times and can insert it into any USB port when required. It contains a microcontroller that runs firmware to control access to personal information and credentials stored on non-volatile memory. It integrates with infrastructure of the Windows operating system to provide multiple levels of security.

The traditional method of providing user authentication is via a username/password combination. You need one set to log on to the network, another to access email, another to digitally sign your email, another to access certain other applications, or files, or data records. Your Information Technology personnel want you to change passwords at least every 90 days and their rules for "acceptable" passwords just gets longer and longer! I, for one, simply cannot remember any more passwords!!!

The hot-plug capability of USB and an identity token such as Rainbow's iKey can solve this problem for us. When prompted for a password we simply attach our personal identity token, we supply an entry phrase and the identity token takes over – it interacts with the operating system and security aware applications to provide correct passwords when required. The operating system will also be alerted when the key is removed. The two-level nature (physical token plus entry phrase) is more robust than a single password and lost keys can be deactivated and replaced. An iKey can store up to 8KB of data including personal information, passwords, cryptographic keys, licenses, credentials, certificates, cookies or other data. This data is encrypted by the USB microcontroller and is saved in non-volatile memory. When the iKey is removed all of your confidential data comes with you and none is left on the hard drive of your PC host.

Rainbow Technologies also provide a Secure Suite application program for Window 2000, which allows Data Encryption and Digital Authentication to be added to any application program or data set. Additionally they provide iKey Software Development Kits with cryptographic API's (some restricted under export laws) so that software developers can develop and integrated solution.

REMOTE DATA COLLECTION

I added an LED and a detector to the block diagram (see Figure 15.2a) and changed the packaging (see Figure 15.2b) to create a portable bar code reader. This example is from Incascan.

Figure 15.2 Miniature bar code reader.

Since this unit operates remotely from the PC host it contains a battery. Bar codes collected throughout the day are uploaded when the unit is connected to the PC host. Once this connection is detected the hot-plug capability of USB, the PC Host invokes a data logging application program which extracts the information from the portable unit then clears its data memory for additional data collection. The battery is also charged while the remote unit is connected.

Do you have a remote data collection application that could be implemented with a similar design?

MORE PORTABLE DATA STORAGE

The block diagram of my next example, and, in fact, the physical appearance shown in Figure 15.3 is similar to the security key example.

Figure 15.3 Miniature "hard drive".

The amount of non-volatile memory has been increased to 64MB and the USB microcontroller firmware has been changed such that this device enumerates as a hard disk drive. The "Moving a Lot of Data" chapter explained how to do this. The Thumb Drive from EiWare is a complete, bus powered, I/O device that the PC host recognizes as a removable hard drive. When connected to the PC host a new drive is added to the system and Windows can copy files to and from it, just like a "normal" hard drive. This is a lot simpler than working with floppy disks since no reader is required and the 40 times increase in density means there are fewer devices to transport. During the development of this book I carried around the whole manuscript, including photographs, on one of these drives!

DATA STORAGE TECHNOLOGIES

The previous example, and several of my following examples were made possible by the recent advances in data storage technologies. I want to spend a few pages looking inside these technologies since their approaches will create new product ideas for many of you.

The floppy disk has been the mainstay for portable data for some time. Its low media cost offset it's other less desirable attributes such as rotating media, high power usage and reader/writer device required. Higher density devices such as the Zip drive from IOMega, the SuperDrive from Imation and the Jaz drive from IOMega provide better storage alternatives but all are based on the same concept of rotating magnetic media.

The first solid-state disk was based on the "floating gate" transistor shown in Figure 15.4.

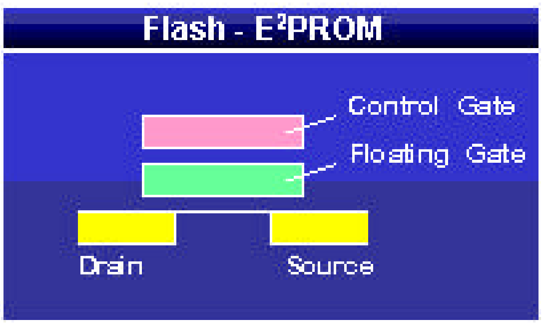

Figure 15.4 A storage transistor.

When there is no charge on the transistor gate the transistor behaves as an open circuit and no current passes between the source and drain connections of the transistor. However, when the gate is charged, it creates a conducting channel between the source and drain connections and it behaves as a closed circuit. We have a digital device that can store a '0' or a '1'. Charge is placed or removed from the floating gate by a control gate. This data storage transistor has many names; EAROM, EEPROM, E2PROM and FLASH (Electrically Alterable Read Only Memory, Electrically Erasable and Programmable Read Only Memory (twice) and a trademark of <<who?>>) and, we shall see in a moment, that even more names are being created for it.

The first EEPROM designs used an internal structure similar to static RAM as shown in Figure 15.5a. The storage transistors were arranged in a XY matrix which allowed any cell to be randomly accessed. Note that each cell requires two transistors, one for access and one for storage. Also shown in Figure 15.5b is an alternate internal arrangement pioneered by Toshiba; this scheme connects many storage transistors in parallel to form a data block.

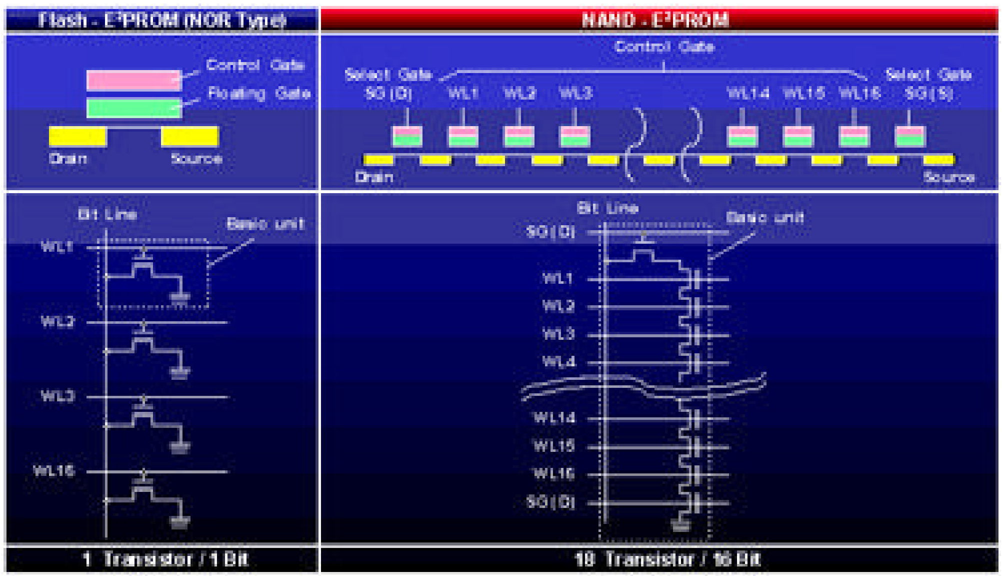

Figure 15.5 Two alternate connection schemes

Only two additional select transistors, one row and one column per block, are required which, in effect, doubles the amount of storage per unit area. The Toshiba scheme does not allow individual byte write capability and is slower at random accesses but is much faster at writes, is much denser and its block orientation makes it an ideal candidate for floppy disk and hard disk replacement.

Toshiba worked with a group of vendors to define a form factor to deliver this high-density data storage product. It was initially called the SSFDC (Solid State Floppy Disk Card) but is better known today by Toshiba's product name of SmartMedia*. Figure 15.6 shows the low-cost packaging (basically a die mounted in plastic) and a typical SmartMedia product. Storage densities ranging from 8 MB to 64MB are available today in products that are about one third the size of a credit card and just as thin.

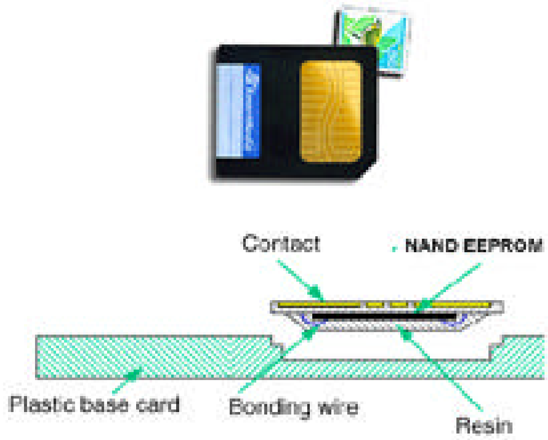

Figure 15.6 SmartMedia products are thin.

An alternate packaging scheme called CompactFlash is shown in Figure 15.7. This format was derived from the popular PCMCIA format but is shorter and uses only 50 of the PCMCIA's 68 signal/power pins.

CompactFlash cards are designed to fit inside the volume envelope of a PCMCIA card. A Type I CF card is only 3.3mm thick while a Type II CF card is 5.5mm thick (the same as a PCMCIA Type II card).

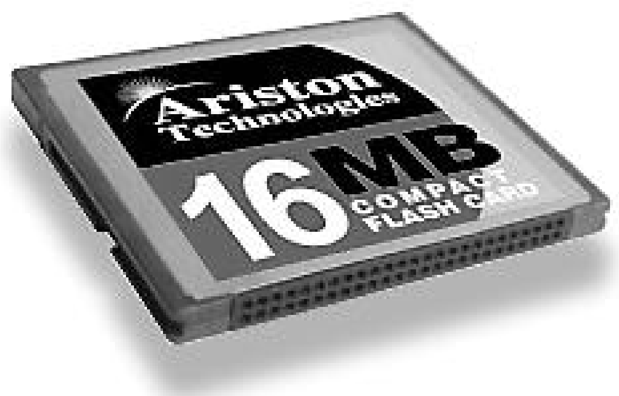

Figure 15.7 CompactFlash cards.

The increased physical volume of a CompactFlash card enables more storage devices to be integrated into a single package. At the time of writing the available products spanned from 8MB to 340MB.

I would have been happy to end this data storage section here – we have replaced floppy disks, and even hard disks, with rugged, compact, solid-state devices. But, a new product from IBM has taken us full circle. IBM took advantage of the CF Type II card format and have mounted a 1-inch diameter hard disk drive and all of the control electronics, inside it. The IBM Microdrive, shown in Figure 15.8 supports much faster data transfer rates than Flash- based cards and is available in 340MB, 750MB and 1GB sizes.

Figure 15.8 IBM's Microdrive enables 1GB in CF Type II format.

ACCESSING DATA STORAGE DEVICES

We have discussed two physical formats of data storage devices – the SmartMedia and the CompactFlash formats. A simple reader device is used to attach these devices to a PC host as shown in Figure 15.9. Note that the block diagram is identical to our third example – the USB controller makes the data storage devices look like attached hard disk drives to the PC host so no special operating system or applications software is required to access them

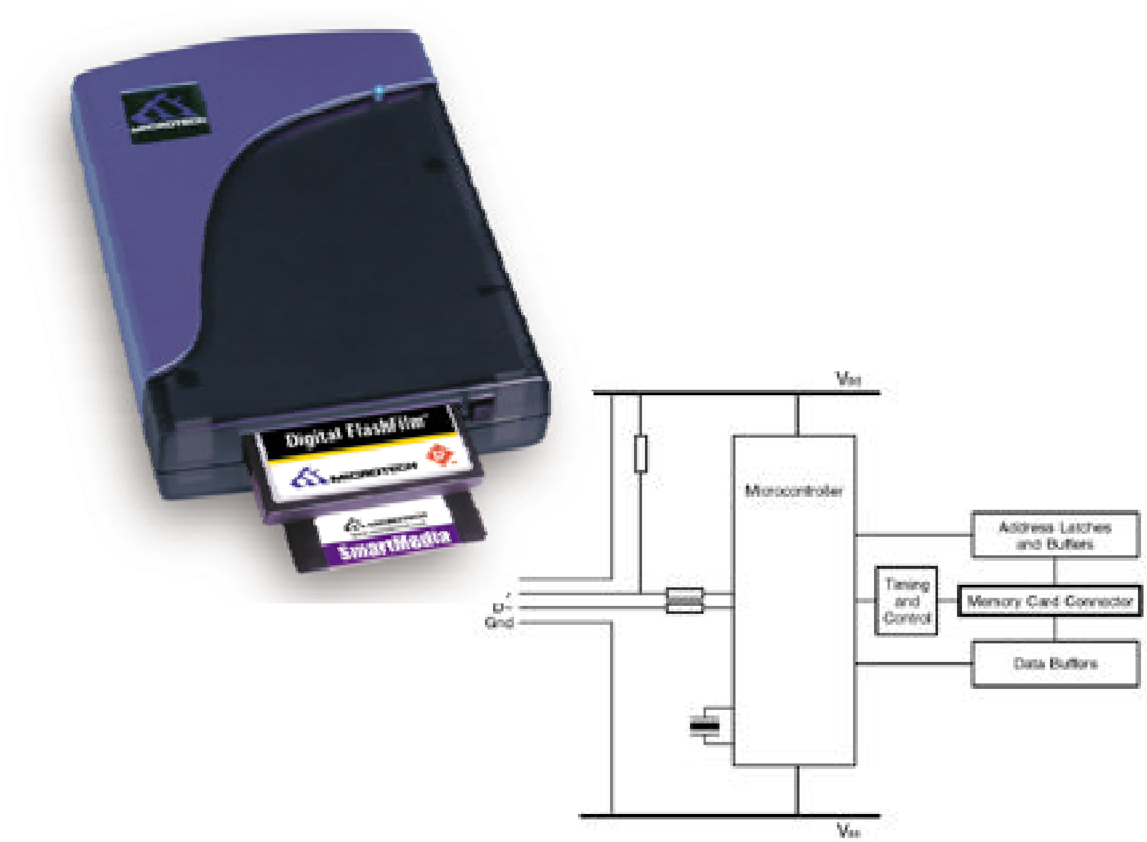

Figure 15.9 Attaching SmartMedia and CompactFlash to a PC host.

So what could we do with 64MB, 340MB or even 1GB of portable data storage? Lets look at several examples.

AN MP3 PLAYER

An MP3 player is a small extension to our “portable hard disk” design as shown in Figure 15.10. The USB microcontroller is also responsible for MP3 decode and an amplifier. The MP3 player is self powered using batteries for full portability.

Figure 15.10 Block Diagram of a MP3 player.

The audio format we dealt with in Chapter 12 “I like the sound of that” was Pulse Code Modulation. This is a simple sampling technique and is used in Windows WAV files. The PCM encoding technique makes the WAV files very large and a 64MB storage device could only hold about 6 minutes of CD quality audio.

MP3 files are compressed audio files and a 10 to 1 compression is common. This means about 60 minutes of audio on a 64MB storage card. This is starting to become a usable product!

MP3 audio compression is a fall-out of the work done by the Motion Picture Experts Group (MPEG) whose overall goal was the compression and synchronization of audio, video and related data streams to an aggregate bit rate of about 1.5Mb/s. This would allow many TV channels to be handled simultaneously via satellites. The MPEG/audio standard can also be used for audio only applications to compress CD quality audio at much lower bit rates.

The basic theory behind MP3 compression is the removal of sounds that the human ear can not hear anyway. An audio CDROM actually contains ten times more information than the human ear and brain can resolve! The MPEG standards committee completed extensive research on the frequency and amplitude response of the human ear and derived a psychoacustic model which is used as part of the compression algorithm. They discovered that the human ear has poor frequency resolving power so a strong audio signal at a fixed frequency will mask weaker audio signals at a similar frequency. Their results showed that the ear's frequency selectivity varies in acuity across the audio frequency spectrum in predefined bands. They concluded that the threshold for noise masking at any given frequency was dependent solely on the signal activity within a critical band of that frequency. The MP3 compression algorithm therefore separates an incoming audio signal into 32 pre-defined bands (time-to-frequency conversion), calculates the dominant frequencies in each band so that the noise threshold can be determined, discards signals that cannot be heard and creates an encoded bit stream as shown in Figure 15.11. The MPEG specification allows for ancillary data, not necessarily related to the audio stream, to be embedded within the encoded bit stream. The decoder simply reverses this formatting and reconstructs the PCM audio stream (s).

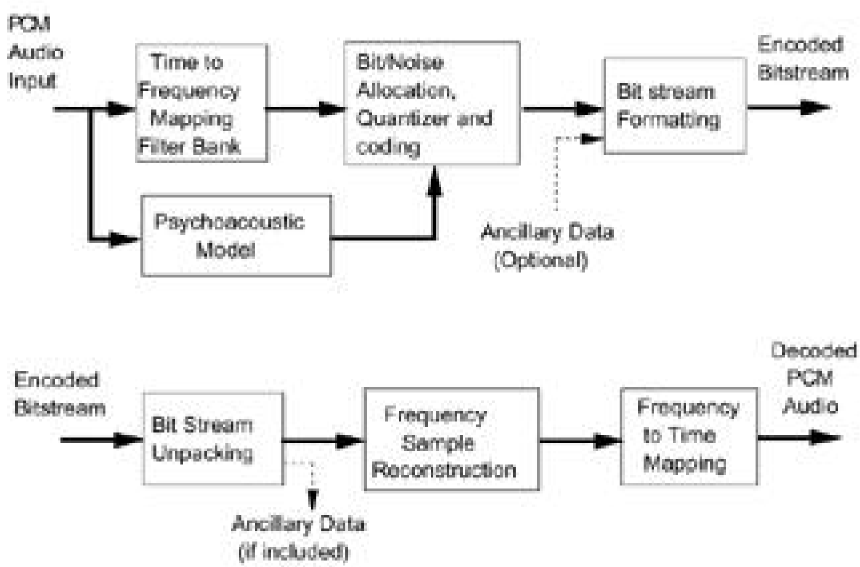

Figure 15.11 MP3 Encoding and Decoding

MP3 encoding to create MP3 files is typically done using software on a PC host. This process is called ripping and is not done in real time due to the extensive computations that must be made. The decoded output quality will directly correlate to the quality of the data encoding so great care is taken during this creation stage.

MP3 decoding must be done in real time else the audio output will be distorted. On a PC host with a Pentium II or III processor this can be completed with less than 3% of the processor. In a portable MP3 player however, a typical 8-bit, USB microcontroller is not powerful enough to run the decode algorithm in real time so a hardware decoder is used as shown in Figure 15.12.

(Full schematics are included on the CDROM)

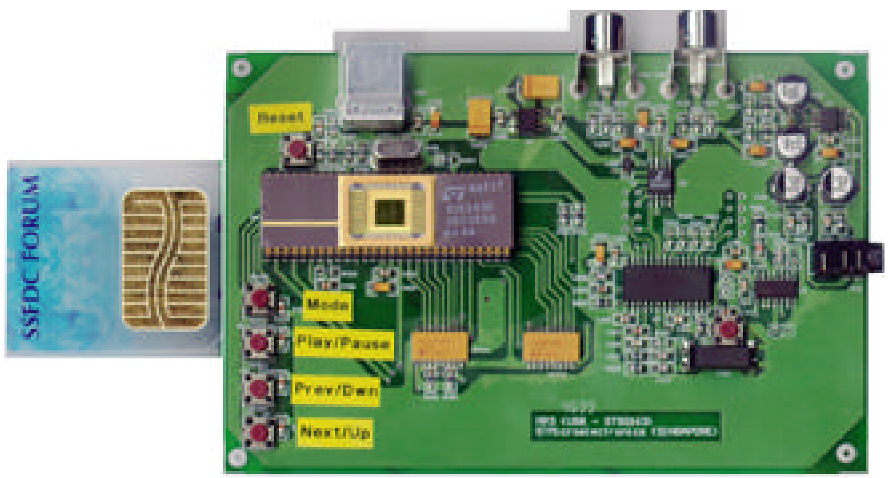

Figure 15.12 MP3 reference design from ST. Microelectronics.

The ST92163 USB microcontroller handles the “hard disk” interface when the unit is attached to the PC host and handles user requests via push buttons when the unit is in portable mode. MP3 files are stored on a SmartMedia card. The MP3 decode is done in hardware using an STA013 decoder – an internal block diagram of this decoder is shown in Figure 15.13. Standard D/A and amplifier are used as the backend. The full source code for this reference design is available from your local ST Microelectronics sales office.

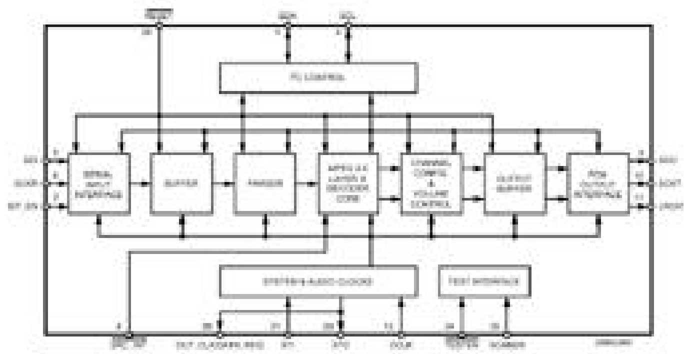

Figure 15.13 MP3 decode in hardware.

An alternate solution from Sigmatel is shown in Figure 15.14. Here the MP3 decode is done in software by a Digital Signal Processor. This DSP also handles the file transfer from the PC host when connected via USB and also controls the interfaces to a variety of data storage devices.

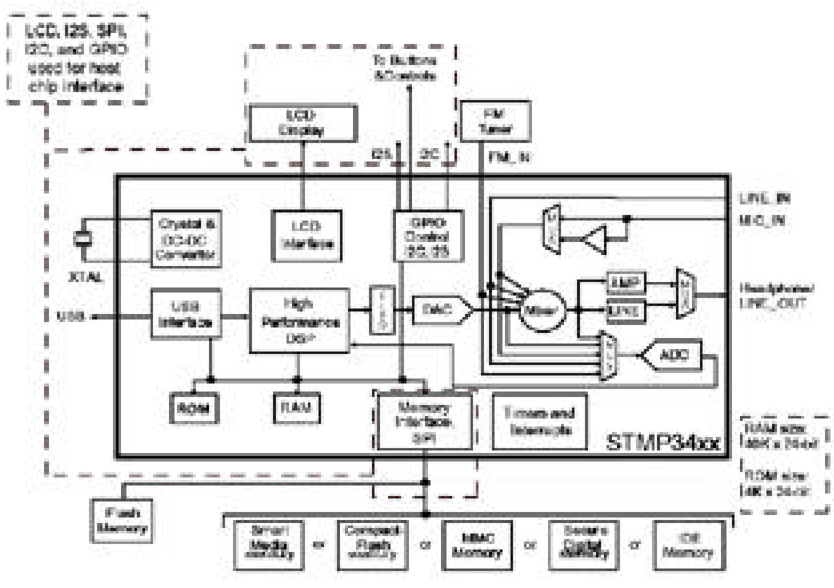

Figure 15.14 MP3 decode in software.

DIGITAL STILL CAMERA

Building a digital still camera is a specialist task due to the large amount of data that must be recorded in a short amount of time. Special purpose hardware must be used and there is a wide range of available products. I shall discuss two solutions; one designed down to sub $50 price point and one designed to include as many features as a consumer is willing to pay for! Both examples operate on battery power and are occasionally connected to the PC host for data transfer. The “front-end” or image sensor is common to both and it is interesting to look at the camera’s similarities before discussing their differences.

Until recently, the only technology available for capturing images in an electronic form was the charge-coupled device or CCD. A CCD sensor requires several voltages and precise clocks to produce a high quality image which is clocked, bit by bit, into adjacent sensing circuits.

In 1993, researchers at NASA’s Jet Propulsion Laboratory invented an image capture method using a standard CMOS technology. This is the same process used to build modern microprocessors, memory components and other standard logic functions. Within a CMOS process, signal sensing and processing can be built into the same die as the sensor. On chip analog-to-digital converters capture the signal early after detection to minimize noise caused by adjacent sensors. This integration reduces chip count and therefore system cost.

The rapid development of the CMOS process by microprocessor manufacturers such as Intel has reduced the geometries of CMOS transistors by a factor of two every 2-3 years (Moore’s law). The CMOS image sensor directly benefits from these reduced geometries

And this has also pushed the competing CCD suppliers to provide higher resolution at lower costs.

The CCD and CMOS sensors detect light in basically the same way. Figure 15-15 shows a single cell of each sensor. Visible light striking the silicon generates electrons which charge up the individual capacitors. A sensor array is made up of a large matrix of identical cells.

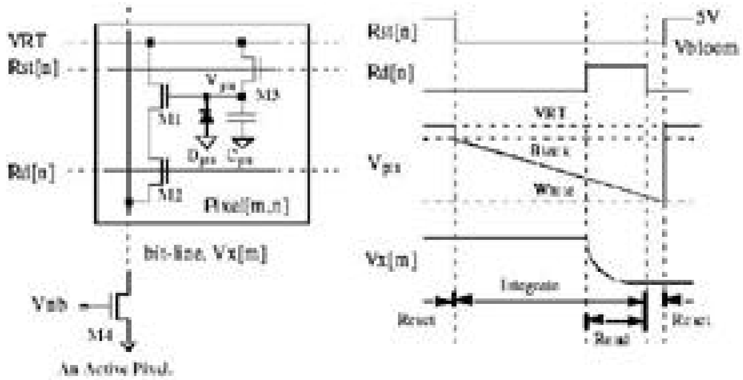

Figure 15-15. Detecting light using CCD and CMOS sensor.

You cannot continually shine light onto these sensors since the individual capacitors will then become fully charged and will leak electrons to adjacent cells (this is called blooming). An electronic shutter is used to expose the cells to the appropriate amount of light – this is similar to a traditional film camera where over-and under-exposure are also undesired effects. The shutter cannot be open too long since movement of the subject or camera will cause blurring of the image. A color sensor is constructed by coating each individual cell with a filter as shown in Figure 15-16. This checkerboard pattern is called a Bayer filter after its inventor.

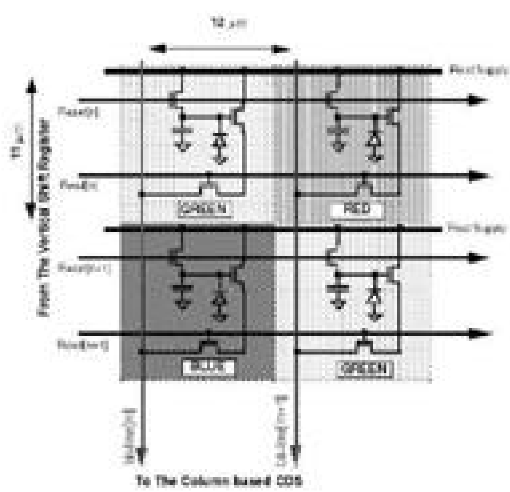

Figure 15-16. Each cell of a color sensor uses four phototransistors.

A CCD sensor will read out the charge on each capacitor by shifting it down a column to an ADC. (Hence the name charge-coupled device). A CMOS sensor, due to it higher integration, could have a buffer amplifier and ADC per cell; the digital signals are easier to strobe out of the sensor array and are not prone to cross-talk errors. The power dissipation of a CMOS sensor is about one tenth that of an equivalent array-size of a CCD sensor with its support circuitry – this is very attractive for our portable digital camera application.

Sensors are specified by their X and Y pixel dimensions. Small sensors (CIF=350x288) are available as well as newer megapixel sensors (1680x1680) from a variety of manufacturers.

LOW COST EXAMPLE

A low cost design will use a small sensor array and this first camera example uses the STV0680+6500 chipset from St. Microelectronics. The block diagram and representative design is shown in Figure 15-17.

Figure 15-17a. Block diagram of low-cost camera example.

Figure 15-17b. Commercial design using this chipset.

Note that this same hardware could be used as a tethered USB video-conferencing camera; this application is presented in Chapter 13.

As a portable camera, this example design needs to store as many images locally as possible. This, of course, means data compression and the STV0680 uses a proprietary compression algorithm (available under license from ST Microelectronics) to reduce the storage requirements of each image. The STV0680 requires fast memory to implement this compression algorithm so the primary memory in this example is SDRAM (the access time of Flash memory produces an unacceptable delay between consecutive images being captured).

Since the SDRAM was required in the design for the compression algorithm, it is also used for long term storage of the images. A single 16Mb SDRAM component can store 20 CIF images while a 64Mb component can store 80 CIF images. A standby mode, which allows full image retention at extremely low system power consumption, gives a battery life of up to several months.

The STV0680 also supports push buttons and switches to control the operation of the camera and an option dual 7-segment display and piezo electric buzzer for user feedback. Typically the 7-segment display is used to display the number of images taken while the piezo electric buzzer produces “success” sounds and “failure” sounds for error conditions. A USB interface is used to transfer images up to the PC host for display, editing and printing.

FULL FEATURED EXAMPLE

My second digital still camera is a third generation COACH (Camera On A CHip) design from Zoran Corporation. The block diagram and a representative design are shown in Figure 15-18.

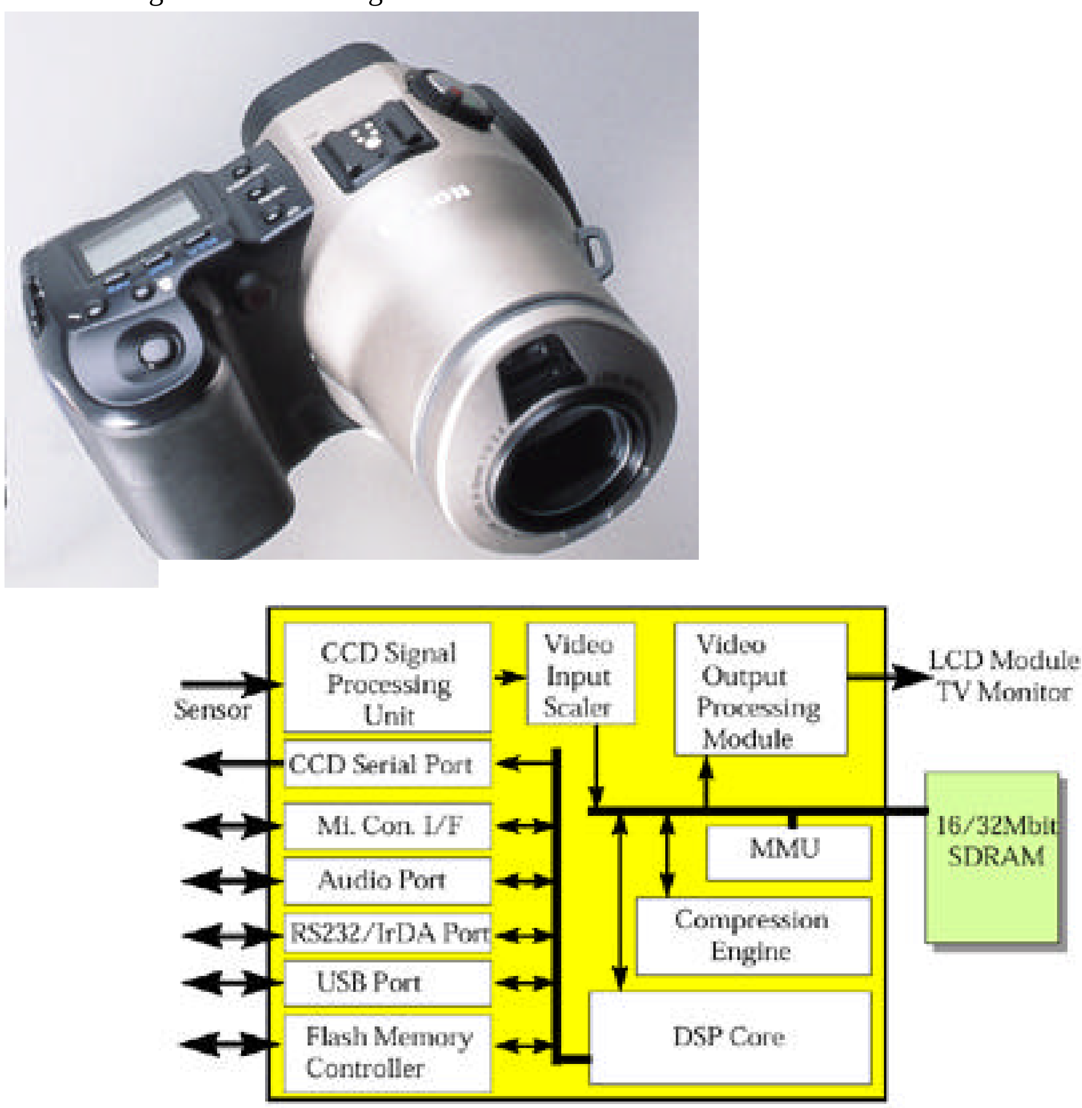

Figure 15-18. Full-featured still camera example.

The COACH component is implemented as a 16-bit Digital Signal Processor with firmware to execute the compression algorithms and other system control functions. Zoran supplies a Camera Profiler tool which allows functions to be extracted from a library to configure firmware for a wide range of camera products. Additionally a separate microcontroller may be used to customize the human interface if required.

A live image is displayed on an eye-LCD display, a view-LCD display or on an attached TV monitor; users like this real time display since it allows them to better compose their image, improve its focus and discard the image if it wasn't exactly as they desired.

A captured image is stored in local SRAM for high speed processing. The COACH component creates a JPEG image in the SRAM and then moves it to the permanent storage of a SmartMedia card or a CompactFlash card. The COACH design allows for the addition of audio to each image; this could be captured during the scene or used for annotation.

The COACH component can also control a Flash Bulb Circuit so that good images may be collected in all light conditions. Captured images may be uploaded to a PC host using a standard USB TWAIN driver or by installing the SmartMedia card or CompactFlash card into a PC host reader. Images can also be downloaded to a COACH camera; if you have to make a remote presentation then this camera is a convenient, low-weight device to transport and playback the images using a standard TV screen.

OTHER REFERENCE DESIGNS

The digital still camera application area is expanding rapidly. Other reference designs are available from Texas Instruments, AOX Incorporated, Kawasaki LSI and others. At the time of writing all devices operated at 12Mb/s. Several 480Mb/s devices are in development but not far enough along to include here. Please refer to the companion web site for the latest information.

PERSONAL DATA ASSISTANT

The examples we have explored in this chapter so far have been fixed function devices. The MP3 player used specialized hardware to play encoded files downloaded from a PC host. The Digital Still Camera used specialized hardware to collect and upload encoded files to the PC host. The bar code reader collected data from the PC host and the hard drive saved data for the PC host.

Rather than use special purpose hardware we could design a portable device with a very high performance microcontroller that implemented the majority of functions in software. A block diagram of a general-purpose reprogrammable I/O device is shown in Figure 15-19. Low cost sensors are still required but all processing the core of this processor operates at only 1.75 volts and has an active power of 600mW. Extensive power management circuitry reduces this to less than 50mW when idle and includes a 50uA sleep state. A portable data assistant built around this technology would have 12 hours or more using a single battery. (A full reference design is available from Intel Corporation).

Figure 15-19. Personal Data Assistant application example.

A device with this much capability would need to have an embedded operating system to manage all of the peripherals and connection to the PC host. The Windows CE operating system is a popular choice. Others are available.

Note that the human interface is different from the PC host. There is no mouse and no keyboard. These have been replaced by touch sensitive screens and voice recognition software. There are also several buttons and a crisp color display. Many PDAs also have an expansion slot so that additional capability may be added – a camera, a pager, a cell-phone or whatever the customer may desire.

The technology of PDAs is still advancing at a rapid pace – it is now possible to build the cell phone/car controller/enemy zapper/finger print scanner/missle launcher that James Bond used in “Tomorrow Never Dies”. As the density of storage continues to increase and the performance of the microcontroller continues to increase, the PDA’s capability will only be limited by your imagination.

The product I am still waiting for is the Smart Comand Recorder. The SCR is a function that uses the microphone, speaker and USB connection of my PDA. At the end of a long day or business trip I am physically tired but still mentally alert. I remember tasks that I should have done or need to do. I pick up my SCR and record “Call Bob, I need the sales figures by 10am,” “Call Little League, team photos are at 5pm after Thursday’s game”, Call Marjorie, the quote for ACME is approved”, “Remind me on Friday to pick up theater tickets”. I empty all of the thoughts chasing around in my head into my SCR.

When I get home I plug my SCR into its USB cradle. My PC host recognizes the SCR and starts to interrogate it for commands. So while I am having dinner and spending quality time with my family my SCR and PC host are implementing all of my commands, picking up phone messages and urgent emails and, of course, recharging the SCR’s battery. I pick up my SCR on my way out to work the following morning and can dispose or action most of my emails and phone messages during the journey to the office.

Imagine how much more productive you would be with a SCR. What if it could also GET/SAVE files such that the data you needed was always on your PDA.

Imagine now what kind of portable device that you could build. All of the essential technology has been described in this chapter to fuel your ideas. The building blocks are in place. What will you build?